

## Listado 1 de Estudio Lenguaje C (OBLIGACIÓN usar funciones)

Al término del desarrollo del listado, mandar al correo: [p.landero@uandresbello.edu](mailto:p.landero@uandresbello.edu) un archivo comprimido con todos los ejercicios resueltos. El **ASUNTO** debe decir: **"Desarrollo Listado 1 Lenguaje C sección \_\_\_\_"**

- 1) Escriba un programa que muestre en la pantalla un mensaje de presentación, pregunte al usuario su nombre y le salude con un mensaje personalizado que use el nombre que se acaba de leer.

**Nota:** La lectura de cadenas de caracteres mediante la función `scanf` se detiene en cuanto se encuentra un espacio en blanco. Por tanto, si al solicitar el nombre se introduce algo como "Arturo de la Vega", solo se almacenará en la variable `cadena` el texto antes del espacio ("Arturo"). Para leer el nombre completo se puede recurrir al especificador de formato `[\n]` que indica que deben leerse todos los caracteres hasta que se introduzca un salto de línea. Por ejemplo `scanf ("%[\n]s",&nombre);`

- 2) Escriba y compile el siguiente programa en C:

```
#include <stdio.h>

int main(void)
{
    int a, b;
    char cadena[8];
    int c;
    a=7; b=14; c=128;
    printf ("Asigne un valor a la cadena de caracteres ");
    scanf ("%[\n]", cadena);
    printf ("La cadena es %s\n", cadena);
    printf ("El valor asignado a las variables
es:\na=%d\nb=%d\nc=%d\n", a, b, c);

    system("PAUSE");
    return (0);
}
```

¿Detecta algún error? Ejecute el programa anterior asignando a `cadena` el valor "Hola", ¿qué sucede? Repita la ejecución asignando a `cadena` el valor "Buenos días", ¿qué observa?

- 3) Escribir un programa en C que escriba los números comprendidos entre 1 y 1000. El programa escribirá en la pantalla los números en grupos de 20, solicitando al usuario si quiere o no continuar visualizando el siguiente grupo de números. Generalizar el programa para que escriba los números comprendidos entre dos valores que introduzca el usuario, y sea éste también quien decida el tamaño del grupo a visualizar por pantalla.
- 4) Diseñe una función de nombre `swap` que, dados dos números flotantes, intercambie sus valores. Los parámetros que debe recibir la función `swap` deben ser punteros. La función `swap` es de tipo `void`.
- 5) Dado un vector de 10 elementos =1, 2, 3, 4, 4, 7, 8, 9, 5, 4, escribir un programa en C (haciendo uso de puntero) que muestre las direcciones de memoria de cada elemento del vector.
- 6) Se quiere averiguar el número mágico de una persona. Para calcularlo se suman todos los números de su fecha de nacimiento y a continuación se reducen a un solo dígito.  
Ejemplo: Fecha de nacimiento: 05/02/1973  $5 + 2 + 1973 = 1980 \Rightarrow 1 + 9 + 8 + 0 = 18 \Rightarrow 9$   
Realice un programa que:
  - a) Solicite al usuario el año, el mes y el día que componen su fecha de nacimiento y compruebe que la fecha introducida es una fecha válida. La fecha se considerará válida si:  $\text{año} > 0$ ;  $1 \leq \text{mes} \leq 12$ ;  $1 \leq \text{dia} \leq \text{DIAS\_MES}$ . Se considerará que, con independencia del año, el número de días del mes de Febrero es 28.

- b) Calcule el número mágico asociado a esa fecha.
  - c) Muestre el número mágico por pantalla
- 7) Realizar un programa que pida al usuario información de un día/mes/año posterior al 1978 y a continuación le muestre al usuario el día de la semana con que se corresponde esa fecha. Para realizar este cálculo la única información disponible es que el 1 de Enero de 1978 fue domingo. Para realizar el ejercicio conviene tener en cuenta que el día de la semana se puede obtener a partir del número de días transcurridos entre la fecha introducida por el usuario y la fecha de referencia (1 de enero de 1978). Una vez conocido este valor, el día de la semana vendrá dado por:  $\text{dia\_semana} = \text{n\_dias\_transcurridos} \% 7$
- 8) Escribir un programa que permita gestionar una máquina expendedora. El programa deberá pedir al usuario el precio del producto que va a comprar y el dinero que ha pagado y calcular la vuelta correspondiente. El programa debe mostrar cuántas monedas de cada tipo deben ser devueltas. Los tipos de monedas a considerar deben ser 1, 5, 10, 50, 100 y 500 pesos, y se asume que hay suficientes monedas de todos los tipos para devolver el cambio. Utilizar funciones para facilitar la legibilidad del programa principal.
- 9) Cree una estructura de nombre Array que contemple las siguientes características:
- a) La creación de un arreglo dinámico.
  - b) El largo del arreglo
- Luego, cree una función de nombre `iniciarArray` que reciba solo una variable de tipo entero, y que retorne una variable de tipo Array que contenga un arreglo de largo `n` con valores aleatorios. Diseñe una función de nombre `mostrarArray` que reciba solo una variable de tipo Array y que muestre el contenido del arreglo.
- 10) Para la gestión de los libros de una pequeña biblioteca es preciso conocer su título, su autor, el ISBN (cadena de 17 caracteres: 978-3-16-148410-0) y si está prestado o no. Escribir un programa en C que:
- a) Defina una estructura denominada `fichaLibro` que permita almacenar los datos de cualquier libro.
  - b) Declarar e inicialice dos variables del tipo `fichaLibro`.
  - c) Compruebe si las variables declaradas representan ejemplares de un mismo libro o de libros distintos.
- 11) Un negocio de paquetería con entregas a domicilio nos ha solicitado un programa para almacenar datos sobre las calles y las viviendas de su Ciudad. Cada calle se identificará por dos elementos: Nombre y viviendas que tiene. Además, cada vivienda estará identificada por el número que ocupa en la calle y la planta. Admitiendo que la ciudad tiene 5 calles y cada calle tiene cinco viviendas, desarrollar un programa que permita al usuario:
- a) Almacenar los datos sobre las calles y las viviendas de la ciudad.
  - b) Ver toda la información sobre una calle: número y viviendas que tiene
- Nota: Por simplicidad se admitirá que los nombres de las calles no contienen espacios.
- 12) Escriba un programa en C que permita almacenar mediante el uso de estructuras los datos de 100 clientes de una empresa de automóviles. El programa permitirá ir almacenando los datos de los clientes, asignándole a cada uno un número según el orden en el que se van grabando. Los datos que se guardarán de cada cliente son el nombre, los apellidos, el teléfono y e-mail. Tras leer un cliente se dará la opción de terminar tecleando un 0. Posteriormente modifique el programa para que evite que un cliente sea guardado dos veces. Para ello se comparará el teléfono con los ya guardados y en caso de que estuviera repetido se mostrará el nombre del cliente, el teléfono repetido y la posición en la que está grabado.
- 13) Implementar en C un programa que maneje un array de estructuras para almacenar las fechas de cumpleaños de 20 amigos. Los datos a almacenar sobre cada amigo son nombre, día, mes y año. El programa debe indicarnos cuántos de nuestros amigos cumplen los años un día y mes determinados introducidos por teclado.